

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 01-251230

(43)Date of publication of application : 06.10.1989

(51)Int.Cl.

G06F 9/06

G06F 12/00

(21)Application number : 63-079362

(71)Applicant : TOSHIBA CORP

(22)Date of filing : 31.03.1988

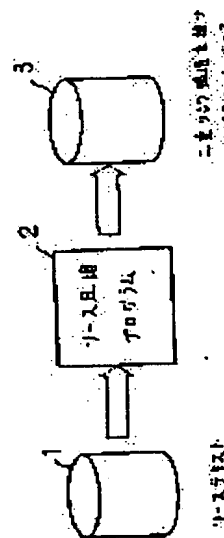
(72)Inventor : YAMATANI YUJI

(54) PROGRAM CONVERTING SYSTEM

(57)Abstract:

PURPOSE: To store line information and a compaction source without waste, and also, to decrease the number of pointers by reading a program from a source text, collecting the data by a block unit and executing a link between its blocks.

CONSTITUTION: A source compression program 2 reads a program of a one-line portion from a source text 1, converts it to a compaction source group by one line unit and a compaction source having a double link structure is stored in a file 3. Line information of plural blocks for holding the compaction source is stored in the end direction from the head and the compaction source is stored in the head direction from the end at every unit for showing one line, and the fore-going is executed until both the data go to unstoreable. The data which cannot be stored in its block A is stored in the next block B. In such a case, the blocks are coupled so that the rear link of the block A points to the block A and the front link of the block B points to the block A. In such a way, a double link structure can be realized by a small number of pointers.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2000 Japan Patent Office

Docket No.
299002053600

⑩ 日本国特許庁(JP)

⑪ 特許出願公開

⑫ 公開特許公報(A)

平1-251230

⑬ Int.Cl.⁴

G 06 F 9/06
12/00

識別記号

4 1 0
3 0 1

庁内整理番号

A-7361-5B
M-8841-5B

⑭ 公開 平成1年(1989)10月6日

審査請求 未請求 請求項の数 1 (全4頁)

⑮ 発明の名称 プログラム変換システム

⑯ 特 願 昭63-79362

⑰ 出 願 昭63(1988)3月31日

⑱ 発 明 者 山 谷 祐 二 東京都青梅市末広町2丁目9番地 株式会社東芝青梅工場
内

⑲ 出 願 人 株 式 会 社 東 芝 神奈川県川崎市幸区堀川町72番地

⑳ 代 理 人 弁 理 士 須 山 佐 一

明 細 書

1. 発明の名称

プログラム変換システム

2. 特許請求の範囲

(1) ソーステキストから1行分のプログラムを読み込み、1行単位でコンパクションソース群に変換する手段と、前記コンパクションソースを保持する複数のブロックのそれぞれに先頭から順に前記変換された1行分のプログラムの行情報を格納するとともに末尾から順にその1行分のコンパクションソースを格納する手段と、前記ブロックのいずれかが一杯になったとき、その一杯になったブロック内に格納された行数と前のブロックを指す前方リンク情報および後のブロックを指す後方リンク情報を格納する手段と、前記ソーステキストを全て読み終えたとき、最初のブロックの前方リンク情報は最後のブロックを、最後のブロックの後方リンク情報は最初のブロックを指すように制御する手段とを備えてなることを特徴とするプログラム変換システム。

3. 発明の詳細な説明

[発明の目的]

(産業上の利用分野)

本発明は、ソーステキストをコンパクションソース群に変換するプログラム変換システムに関する。

(従来の技術)

ソーステキストをコンパクションソース群に変換するシステムが随々提案されているが、従来のシステムは、例えばリスト処理におけるように、ソーステキストのプログラムを1行単位に、データであればワード単位に前後のチェーンで接続する方式が主流であった。

(発明が解決しようとする課題)

しかしながら従来のシステムによると、行情報とコンパクションソースのメモリ空間に無駄が生じ、ポインタの数が多くなるという問題がある。

本発明はこのような事情によりなされたもので、行情報とコンパクションソースを無駄無く格納することができ、ポインタの数も少なくなるプログ

ラム変換システムの提供を目的としている。

〔発明の構成〕

(課題を解決するための手段)

本発明のプログラム変換システムはこの目的を実現するべく、ソーステキストから1行分のプログラムを読み込み、1行単位でコンパクションソース群に変換する手段と、前記コンパクションソースを保持する複数のブロックのそれぞれに先頭から順に前記変換された1行分のプログラムの行情報を格納するとともに末尾から順にその1行分のコンパクションソースを格納する手段と、前記ブロックのいずれかが一杯になったとき、その一杯になったブロック内に格納された行数と前のブロックを指す前方リンク情報および後のブロックを指す後方リンク情報を格納する手段と、前記ソーステキストを全て読み終えたとき、最初のブロックの前方リンク情報は最後のブロックを、最後のブロックの後方リンク情報は最初のブロックを指すように制御する手段とを備えている。

(作 用)

1バイトごとに別々に処理されるものと仮定する。

数字は符号付きの2進数で表現されるものと仮定する。したがって1バイトでは-128~+127までの表現が可能である。

さらにブロックを格納するファイル構成は番号の指定によって各ブロックのアクセスが可能な構成にされていると仮定する。

本実施例のシステムは、ソーステキストを入力し、第2図に示す形式の複数のブロックに変換後、それを出力する機能を有する。

第3図に示すように出力される各ブロックは、前方リンク(フロントリンク)および後方リンク(バックリンク)の両方向のブロックへのポインタを有している。そして1番最初のブロックの前方リンクは1番最後のブロック、1番最後のブロックの後方リンクは1番前のブロックを指す。

一般に、ソースプログラムはインデントーションおよびプログラム自体の見易さのために空白等が多く存在することが多い。そこで第4図に示すように、入力となるソーステキストは1行ごとに

本発明のプログラム変換システムではブロック、すなわちある程度まとまった単位にプログラムを集め、そのブロック間でリンクを行なうことにより、少ないポイントで2重リンク構造を実現する。

また決まった大きさのブロックにプログラムを格納することにより、メモリ空間に無駄が生じない。

(実施例)

以下、本発明の実施例の詳細を図面に基いて説明する。

第1図は本発明の一実施例の構成を示す図である。

同図において1はソーステキストが格納されているファイル、2はソースプログラムの圧縮を行なうプログラム、3は二重リンク構造を持つコンパクションソースを格納するファイルである。

そして本実施例では、1文字が1バイトから表現されると仮定する。また漢字1文字は2バイトからなるものとし、内部的には2文字として扱い、

3種類(A、B、C)の形式にコンパクションを行う。

A形式は先頭1バイトが16進で“FF”、次の1バイトが長さを表わす2進数の計2バイトからなる。

B形式は先頭1バイトが16進で“FE”、次の1バイトが長さを表わす2進数、その次の1バイトが文字を表わす計3バイトからなる。

C形式は先頭1バイトが長さを表わす2進数、次がその長さ分の文字列からなる。

そしてソーステキストの元のイメージを保つために以下のような規則でコンパクションを行なう。

①空白が2個以上続いた場合は、A形式にする。

②空白以外で同じ文字が3個以上続いた場合は、B形式にする。

③その他の場合には、C形式にする。

出力となる1つのブロックの構造は、第2図に示したように固定情報と可変情報とに分けられる。

固定情報は前方リンク、後方リンクおよびそのブロックに格納されている行数が各2バイトずつ

の計6バイトからなり、固定的にブロックの頭部にとられる。

可変情報は行情報部とコンパクションソース部との2つに分けられる。

行情報部は、そのブロックに格納された行数分（固定情報部の格納行数と等しい）の行情報が存在する。1つの行情報はその行を表わす型および長さが各1バイト、そのブロック内の位置（先頭を1とした相対位置）およびソーステキスト全体の順番を示す文番号が各2バイトの計6バイトからなる。

コンパクションソース情報部には、各行情報に対して（ソーステキストの最後を示す型を持つものを除き）、第3図に示した形式のコンパクションソース（1行に対し複数個となる）が格納されている。

最後にブロック内に無駄無く各データを格納するために、第5図に示すように行情報は上（先頭）から下（末尾）方向へ、コンパクションソースは1行を表す単位ごとに下（末尾）から上（先頭）

方向に格納してゆき、これを両方のデータが共に格納不能になるまで行う。そのブロック（ブロックAとする）に格納できなかった両方のデータは、次のブロック（ブロックBとする）に格納する。

また、このときにブロックAの後方リンクはブロックBを指し、ブロックBの前方リンクはブロックAを指すようにブロック間の結合を行なう。

なお、プログラムが持つ附随的なデータ（ソースの名前、作成日付、更新日付など）は、他の固定的なブロックが持つものとする。すなわちソーステキストを他のユーティリティ等で扱いよくするために、物理的に変換する機能をもつ。

かくして本実施例によれば、ブロック単位にデータを集め、そのブロック間でリンクを行なうので、少ないポインタで二重リンク構造を実現することができる。

また決った大きさのブロックにデータを格納するので、メモリ空間に無駄が生じない。

さらにソースプログラムの先頭から末尾をアクセスする、または末尾から先頭をアクセスする等

制御が極めて容易である。

なお本実施例の機能をコンパイラに組込んで、変換されたソースプログラムと出力されたオブジェクトとを同時に出力するようにすれば、オブジェクトと対応するソースプログラムが必ず存在することになるので、デバッグ処理に使用することができるようになる。

【発明の効果】

以上説明したように本発明のプログラム変換システムは、ブロック単位にデータを集め、そのブロック間でリンクを行なうので、行情報とコンパクションソースを無駄無く格納することができ、ポインタの数も少なくなる。

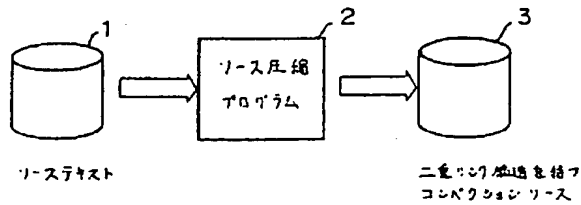
4. 図面の簡単な説明

第1図は本発明の一実施例の構成を示すブロック図、第2図は同実施例におけるソーステキストのブロックを説明する図、第3図はソーステキストのコンパクションを説明する図、第4図はソーステキストのブロック間のリンクを説明する図、第5図はブロック内への行情報およびコンパクシ

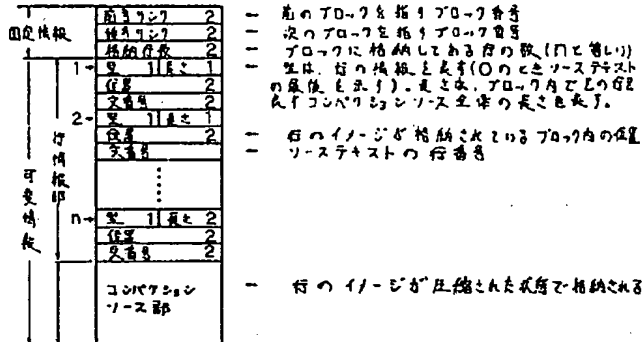
ョンソースの格納方法を示す図である。

- 1…ソーステキストが格納されているファイル、
- 2…ソースプログラムの圧縮を行なうプログラム、
- 3…二重リンク構造を持つコンパクションソースを格納するファイル。

出願人 株式会社 東芝
代理人 弁理士 須山 佐一



第 1 図



第 2 図

- A 形式

FF	L
----	---

 Lバイトの空白
- B 形式

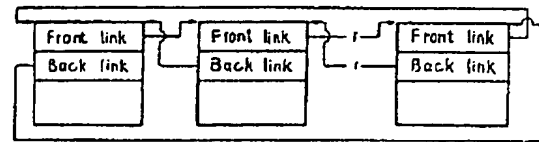
FE	L	C
----	---	---

 1バイト文字CがLバイト
- C 形式

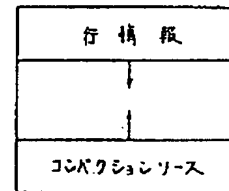
L	image
---	-------

 Lバイトの文字イメージ(L<127)

第 3 図



第 4 図



第 5 図